

## STRUCTURAL COMPUTER-AIDED DESIGN OF CURRENT-MODE CMOS LOGIC CIRCUITS.

Siep Onneweer<sup>\*</sup>, Hans Kerkhoff<sup>\*</sup> and Jon Butler<sup>\*\*</sup>

<sup>\*</sup>Twente University, Enschede, the Netherlands

<sup>\*\*</sup>Naval Postgraduate School, Monterey, CA, USA.

### Abstract.

A set of CAD tools for the synthesis and layout generation of multiple-valued current-mode CMOS logic (CMCL) circuits is described. The synthesis method is based upon the cost-table method. The general circuit structure, the cost-table functions and the decomposition procedure used in the cost-table synthesis program are explained. The synthesis program is based upon a logically complete set of basic elements for CMCL circuits. After circuit synthesis the actual layout is generated using standard-cell IC design tools.

### 1 Introduction.

Recently, the implementation of multiple-valued logic based on Current-Mode CMOS circuits has received much attention in literature [1-7]. Current-mode CMOS logic appears to be attractive for two reasons. First, the IC process technology is the same as that of binary CMOS integrated circuits. This opens the possibility for integrating multiple-valued modules realized in CMCL with standard binary logic circuits in a single VLSI system. The second reason is the powerful logic functionality of CMCL circuitry, which has been convincingly demonstrated by recent developments [7].

However, no publications have yet appeared on computer-aided design tools for current-mode CMOS logic (CMCL) circuits. The availability of these CAD tools, freeing the logic designer from the details of electronics and layout of CMCL circuits, is essential for the acceptance of these circuits.

This paper presents a study and resulting implementation of a design tool for CMCL circuits. The set of tools consists of two parts: a logic-synthesis program and a layout-generation part.

The logic-synthesis program is based on the cost-table approach [8-12]. The function to be synthesized, the target function, is decomposed into a number of functions that are present in a so-called cost table. The cost table contains a logically complete subset of all possible functions to be synthesized. Each of the functions in the table has a certain cost. The synthesis procedure generates the decomposition of the target function into cost-table functions in such a way, that the

resulting implementation has the lowest possible total cost.

The layout of each of the cost-table functions is stored in a library. Standard-cell placement and routing tools are used to generate the layouts of the target functions. The resulting layout is a module that can be used in a VLSI design environment as part of a digital I.C.

At present the system can be used for one-input functions only. An extension to multiple-input functions will be the subject of future research.

Now, an overview of the subjects treated in each of the following chapters will be given. In chapter 2, current-mode CMOS circuits will be introduced. A set of basic elements is described from which CMCL circuits can be built. For these elements logic symbols for use in logic diagrams (in contrast to electronic diagrams) are proposed. It is also shown how the elements can be used to compose simple CMCL circuits.

Chapter 3 gives an overview of design methods for multiple-valued functions. A distinction is made between mathematical and technology-linked methods. Their significance for CMCL circuit design is discussed and the choice for using the cost-table technique approach, will be argued.

The cost-table design method for one-input four-valued CMCL circuits is treated in chapter 4. After comparison of the features of CCD circuits (for which the cost-table method was originally proposed [8]) and CMCL circuits, the general circuit structure and its major constituent parts are discussed. The decomposition procedure, which has been implemented as a computer program, is designed to utilize the full potential of this structure.

Finally, in chapter 5 the conclusions from this work are presented and some suggestions for further development of the design system made.

### 2. Basic circuit elements for current-mode CMOS logic circuits.

#### 2.1. Introduction.

In this chapter, the set of basic circuit

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE <b>MAY 1988</b>		2. REPORT TYPE		3. DATES COVERED	
4. TITLE AND SUBTITLE <b>Structural Computer-Aided Design of Current-Mode CMOS Logic Circuits</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>Naval Postgraduate School, Department of Electrical and Computer Engineering, Monterey, CA, 93943</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited.</b>					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <b>A set of CAD tools for the synthesis and layout generation of multiple-valued current-mode CMOS logic (CMCL) circuits is described. The synthesis method is based upon the cost-table method. The general circuit structure, the cost-table functions and the decomposition procedure used in the cost-table synthesis program are explained. The synthesis program is based upon a logically complete set of basic elements for CMCL circuits. After circuit synthesis the actual layout is generated using standard-cell IC design tools.</b>					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES <b>10</b>	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

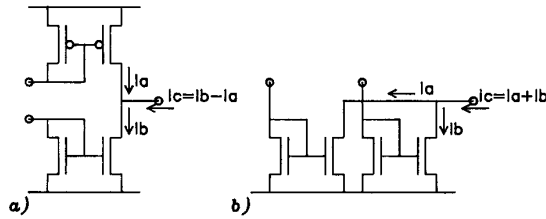


Fig. 1. The two situations occurring if current-mirror outputs are interconnected. a) opposite-type mirrors result in subtraction. b) two mirrors of the same type result in addition.

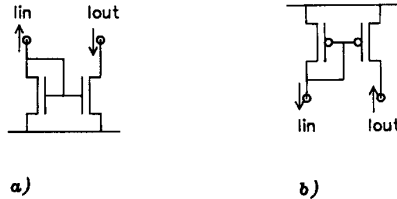


Fig. 2. Definition of current references for the mirrors: a) for N-type mirrors; b) for P-type mirrors.

elements suitable for logic decomposition and logic design of current-mode CMOS logic circuits, is treated. For each element, the logic operation is described, a logic symbol proposed and the circuit realization given.

Analyzing a large number of manually-designed current-mode CMOS logic circuits, [2-7] it turns out that only a few basic elements make up most of the circuits. From these, the most elementary ones have been selected. The resulting set of basic circuit elements is presented. The selected elements are simple, constitute a logically complete set of basic elements and operate completely in the current domain.

In section 2.2, the properties of the logic symbols are explained, and the basic circuit elements are presented in section 2.3.

## 2.2. Logic diagrams of CMCL circuits.

In this paper, logic diagrams instead of electronic circuit diagrams are mostly used in order to formalize the design procedure. In the transfer from electronic circuit diagrams to logic diagrams, one encounters a problem with the signs of signals.

Every branch in an electronic circuit has a current reference direction, which can be indicated by an arrow in the diagram. In most previous publications on current-mode logic, all currents represent a positive logic value. In the translation from a circuit diagram into a logic diagram, confusion occurs regarding the signs of the currents and logic signals. For instance, if the outputs of two mirrors of opposite types are connected, the output currents (fig. 1.a) are

subtracted. However, if the two mirrors are of the same type, the operation represents an addition (fig. 1.b).

In this paper, a current reference direction is defined to avoid this confusion, as illustrated for current mirrors in figure 2. By this definition, the output current of an N-type mirror is always positive, and that of a P-type mirror is negative.

In a logic diagram, the connections between components indicate the direction of the information transfer, not of the corresponding (physical) current. The direction of a physical current must be represented by attaching a sign to the corresponding logic signal. Negative logic values will be indicated by an underscore (e.g.  $\underline{1}$ ). The minus sign was avoided because of possible confusion with the operations subtraction and sign inversion.

## 2.3. Explanation of the basic elements.

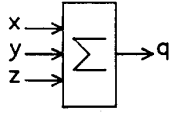
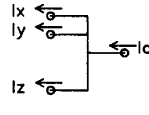
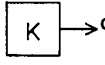
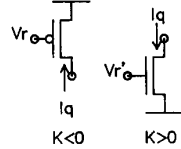
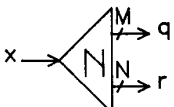
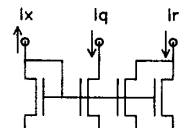
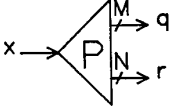
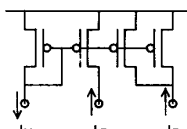
The proposed set of basic circuit elements is listed in Table I. For each basic element, the logic symbol, a mathematical description of the operation and the circuit realization are given in this table.

**Summation.** The simplest operation in the current domain is the arithmetic addition of signals, for which the summator is proposed. If the result  $q$  is outside the range  $0..(R-1)$ , no truncation or other operation (like folding or modulo- $R$  addition) takes place. The summator allows no fanout; it has one output that can be connected to one other circuit element only, usually a mirror. The summator is the only connective operator, which combines two or more signals, in the set of basic circuit elements. The circuit realization of the summator is an electrical interconnection between all the inputs and the output.

**Constant value.** The second basic element is the constant generator. It can provide any positive or negative integer value. In practice, the values used will be small, e.g. between  $-(R-1)$  and  $(R-1)$  (where  $R$  denotes the radix). The corresponding electrical circuit is a constant current generator, consisting of one or more MOS transistors. A positive constant is generated by (an) N-channel transistor(s) and a negative constant by (a) P-channel transistor(s), as indicated in the table. The gate(s) of the transistor(s) are connected to a reference voltage, which can be generated locally [4].

**N-type and P-type mirrors.** There are two kinds of mirrors: an N-type and a P-type. The mirrors are one-input, multiple output logic operators. A mirror truncates its input signal (according to the type of the mirror) and inverts its sign, as is illustrated in figure 3. In the logic description of the mirrors, the mathematical functions  $bndp$  (bound-positive) and  $bndn$  (bound-negative) have been used. They are defined as:

$$\begin{aligned} bndp(x) &= \max(0, x) = 0 \vee x \\ bndn(x) &= \min(0, x) = 0 \wedge x \end{aligned} \quad (1)$$

Table I. Basic CMCL Elements.			
NAME	LOGIC OPERATION	SYMBOL	CIRCUIT REALIZATION
Arithmetic Summator	$q = x + y + \dots + z$		
Constant	$q = K$		
"N"-Mirror	$x' = \text{bndn}(x)$ $q = -M \cdot x'$ $r = -N \cdot x'$		
"P"-Mirror	$x' = \text{bndp}(x)$ $q = -M \cdot x'$ $r = -N \cdot x'$		

The circuit realizations of the mirrors are the well-known N-channel respectively P-channel current mirrors [4], as shown in Table I. Multiple copies of the output signals are obtained by the use of a number of separate output transistors. Multiplication of the output signals with an integer factor is obtained by connecting the drains of several identical output transistors together.

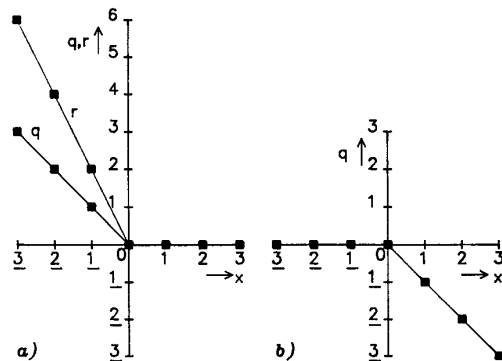


Fig. 3. The logic behavior of the mirrors. a) 2-output N-type mirror;  $M = 1$ ,  $N = 2$ . b) 1-output P-type mirror;  $M = 1$ .

#### 2.4. The necessity of logic level restoration.

The functions obtained with these CMCL basic circuit elements are not level-restoring. They may be viewed as analog functions consisting of linear segments, that have the desired values at the discrete points allowed in the multiple-valued logic system. After a few of these CMCL circuits, the logic levels have to be restored, because of the generation and propagation of deviations from the exact logic values. A level restoration circuit can be found in [5]. This noise margin problem imposes a very important restriction on the use of CMCL circuits, and will have to be investigated thoroughly.

#### 3. The applicability of existing synthesis techniques for CMCL.

In this chapter, some published methods for the synthesis of MVL functions will be briefly reviewed. Based on the structural resemblance between CMCL and some other forms of MVL circuits (IIL and CCD), the direction in which the investigations have been directed will be discussed. In the next chapter, one of the reviewed synthesis methods, the so called cost-table synthesis method, will be treated in detail, and subsequently applied to CMCL.

Commonly, two classes of synthesis techniques

### 3.1. Mathematical synthesis methods.

In contrast, the algebra's developed by Allen and Givone [15], by Vranesic [16] and by Su and Sarris [17] embody elementary operators more suitable for implementation in electronic circuits. The Allen-Givone algebra uses unary literal operators and the maximum and minimum connectives. It can be shown that these operators can be implemented using the basic elements of CMCL from table I. In this way, the functional completeness is easily proved. Some of the literals, however, are relatively costly to implement and the same is true for maximum and minimum circuits in the case of many inputs.

Fig. 4. Example of a synthesis technique by linear combination of the basis functions  $v_0 = \langle 0 \ 1 \ 2 \ 3 \rangle$ ,  $v_1 = \langle 0 \ 0 \ 1 \ 2 \rangle$ ,  $v_2 = \langle 0 \ 0 \ 0 \ 1 \rangle$  and  $v_3 = \langle 1 \ 1 \ 1 \ 1 \rangle$ :  $f(x) = \langle 1 \ 0 \ 1 \ 0 \rangle$ .

Technology-linked methods have been published for IIL and CCD technology. The resemblance between the basic operators in these technologies and those of CMCL, make it likely that an adaptation of these methods for current-mode CMOS logic is possible.

figure 4 the logic diagram for the function  $f = \langle 1 \ 0 \ 1 \ 0 \rangle$  is shown. Essential for this method is the availability of subtraction, which is easily achieved in CMCL. The synthesis procedure can easily be implemented in a CAD program.

#### 4. The cost-table decomposition method applied to CMCL circuits.

#### 4.1. Comparison between CCD and CMCL circuits.

- Within the CCD circuit, copying of charge and crossing of signal lines must be avoided as much as possible. However, at the input, signal copying can occur freely, as the input signal is a voltage.

- Only positive charge packets are available
- The subtraction of signals is expensive in terms of circuit cost [12].

For current-mode CMOS circuits, the freedom is much greater. The most important advantages over CCD technology are:

- Copying signals and multiplication with a constant are possible inside a CMCL circuit, because extra output transistors can be added to mirrors.
- Signal lines are allowed to cross freely without causing interference.
- The signals can be both positive and negative.
- Sign inversion is available and therefore subtraction as well as addition can be used.

On the other hand, CMCL circuits also have some restrictions due to their functioning in the current domain:

- The input signal is a current and can not be applied directly to all subcircuits. It must be distributed by a dedicated subcircuit.
- The fanout is one and can only be increased if an output mirror is used.

From these properties, the conclusion can be drawn that the cost-table method is applicable to CMCL but has to be modified.

#### 4.2. The circuit structure for cost-table synthesis of CMCL circuits.

The cost-table synthesis method is based on a fixed circuit structure for the target function, which is determined by the specific IC technology used. As mentioned previously, the properties of CMCL are different from those of CCD's. Still, the nature of the cost-table method requires a fixed circuit structure.

The proposed general circuit structure for cost-table synthesis of unary CMCL functions is shown in figure 5. Similar structures have often been used in manual designs. Three parts can be distinguished: the input distribution circuit (A), the output combination circuit (C) and the actual logic part which is composed of cost-table functions ( $B_1 \dots B_n$ ). These parts will be discussed in detail in the next three subsections.

##### 4.2.1. The input signal distribution.

In fig. 6, two possible circuits for the distribution of the input signal are shown. A positive input signal according to the definitions in section 2.2 has been assumed.

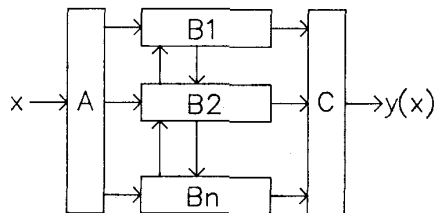


Fig. 5. The general structure of a one-input CMCL circuit for cost-table synthesis. A: input signal distribution circuit.  $B_i$ : cost-table circuits. C: output combining circuit.

**A P-type mirror.** The most simple circuit that can be used to distribute the input signal is the P-type mirror (fig. 6.a). By examination of manually designed CMCL circuits, it was found that this structure is often used. Besides providing multiple copies of the input signal, the mirror inverts the sign. Instead of the signal  $x$ , the negative signal  $\bar{x}$  is applied to the subcircuits.

**An N-type complement circuit.** A second possible input distribution circuit for positive input signals is a complement circuit employing an N-type mirror (fig. 6 b). The distributed signal in this case is the complement  $\bar{x}$ . The complement circuit is slightly more complicated than the P-type mirror. On the other hand, its delay time is considerably less [5]. The types of realizable cost-table functions are the same in both cases. Only the values of the constants are higher for cost-table circuits designed for the complementing input, causing their cost and the power consumption to be higher.

The choice of the input circuit is a tradeoff between circuit complexity, power dissipation and switching speed. For the reasons stated above, the P-type mirror will be used to distribute the input signals.

##### 4.2.2. The output circuit.

If the target function is a cost-table function (or a multiple thereof), no output circuit is needed. In the other case, the output circuit is necessary to combine the cost-table functions resulting from the decomposition procedure. Two operations will be considered for this purpose: the summation and the summation with complement. Other candidates, e.g. maximum, minimum, or inhibit are more expensive in terms of circuit realization, especially for three or more inputs.

**Summation.** The combining operation most natural for current mode circuits and the only one appearing in the table of basic circuit elements (Table I) is the summator (fig 7.a). It does not require any active elements. However, it has one major drawback: it can not provide a fanout larger than one. If multiple outputs are required, each of the cost-table functions must be equipped with additional outputs as indicated in figure 7.b.

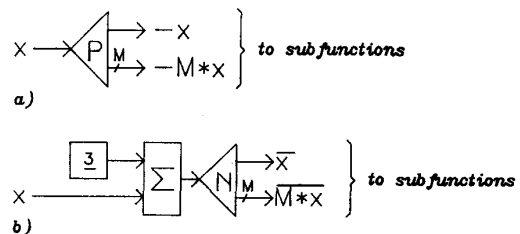


Fig. 6. Two potential input distribution circuits. a) A P-type mirror. The distributed signal is  $\bar{x}$ . b) A complement circuit. The distributed signal is  $\bar{x}$ .

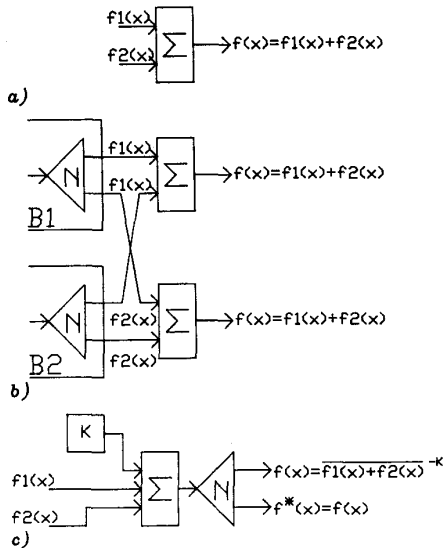


Fig. 7. Output combining circuits. a) Summation without fanout. b) Summation with fanout. c) Complement circuit with fanout.

**Summation followed by a complement.** Alternatively, the fanout can be provided by means of a mirror (N-type for a positive output signal) as shown in fig. 7.c. The output mirror transforms the output signal. It inverts the sign, truncates the signal and optionally multiplies it with a constant integer factor. A negative constant must be added to the (positive) cost-table functions to obtain the proper sign for the input signal of the N-type output mirror. The resulting transformation is the complement operation:

$$f(x) = -\text{bndn}(K + f_1(x) + \dots + f_n(x)) \quad (2)$$

where  $K < 0$  and  $f_1(x) \dots f_n(x)$  are the cost-table functions. The transformation can be usefully exploited in the decomposition procedure, which is treated in section 4.3.

The output circuit proposed in this paper is that of figure 7.c, because of its fanout facility and the additional logic transformation mentioned in the last paragraph.

#### 4.2.3. The cost-table functions.

In reference [8] the term cost indicates the relative cost of the basic gate configurations for digital CCD's. Such diverse properties as chip area consumption, the number of power-supply and data lines, and sensitivity to process and voltage variations were combined in a single cost measure. In later publications the term cost was also used to indicate mathematical properties of cost-table functions. Some of these properties of functions were shown to correlate with their cost of realization as a CCD circuit [10]. The mathematical cost functions used by Schueller et al. are the transition count, total transition size, constant and sum [10].

For a CMCL circuit, its number of (unit) transistors is a natural cost measure derived directly from the electronic implementation. This transistor count  $Q$  has a strong relation with the chip area and the power dissipation. A cost function is useful to guide the decomposition procedure. The correlation between  $Q$  and some of the mentioned mathematical cost functions will be investigated, after the introduction of the cost-table circuits and functions.

The cost table consists of a subset of all 256 four-valued functions of one variable. It must fulfill the following requirements:

- The cost-table functions must be positive. This is required by the synthesis procedure. It implies an ordering of the cost-table functions, in the sense that if  $f_{a+b}$  is the sum of two functions  $f_a$  and  $f_b$  then  $f_{a+b}(x) \geq f_a(x)$  and  $f_{a+b}(x) \geq f_b(x)$  for any value of  $x$ .
- The set of cost-table functions must be logically complete: all functions must be decomposable into cost-table functions. This requirement implies that the cost table must contain the unity delta literals  $\delta_0 = \langle 1 \ 0 \ 0 \ 0 \rangle$ ,  $\delta_1 = \langle 0 \ 1 \ 0 \ 0 \rangle$ ,  $\delta_2 = \langle 0 \ 0 \ 1 \ 0 \rangle$  and  $\delta_3 = \langle 0 \ 0 \ 0 \ 1 \rangle$  because these can never be decomposed.
- For every cost-table function, the realization with the smallest cost (transistor count) must be provided.
- No cost-table function must be decomposable into other cost-table functions.

The properties of CMCL and of the complementing output structure also have consequences for the cost-table functions. Only functions of which the minimum logic value is 0 and the next-to-lowest value is 1 have been included in the cost table. The outputs of the cost-table circuits are mirrors (see the description in the following paragraphs). This makes implementation of a constant multiplication factor easy. The resulting function will be truncated by the output circuit so it is no problem if its maximum value exceeds  $(R-1)$  because of the multiplication.

A small number of circuit structures can be employed to generate many different logic functions by variation of the circuit parameters. N-type and P-type mirrors are both available in CMOS. However, P-type mirrors are 2 to 3 times slower than the N-type mirrors. Therefore, only N-type mirrors will be considered.

Experience with hand-made designs has guided the choice of the cost table circuits. A recurring subcircuit is the combination of a constant, a summator and a mirror (as shown in fig. 8). Several configurations of one or more of these subcircuits have been simulated with a range of parameter settings. Of the functions obtained, only those were included in the cost table that meet the requirements as stated above.

**circuit structure I.** The first structure is shown in fig. 8.a. In fig. 8.b the logic behavior is depicted for some values of the three parameters  $a$ ,

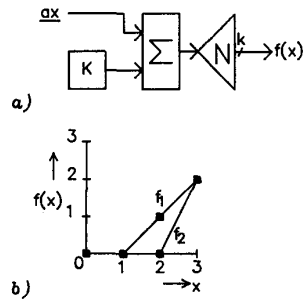


Fig. 8. Cost-table circuit structure I. a) Logic diagram. The parameters are  $a, K$  and  $k$ . b) Some realized functions:

$$f_1(x) = \langle 0 \ 0 \ 1 \ 2 \rangle, f_2(x) = \langle 0 \ 0 \ 0 \ 2 \rangle$$

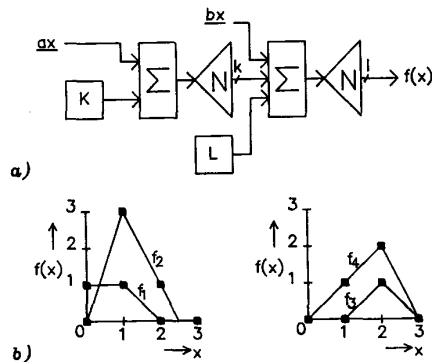


Fig. 9. Cost-table circuit structure II. a) Logic diagram. The parameters are  $a, b, K, L, k$  and  $l$ . b) Some realized functions:  $f_1(x) = \langle 1 \ 1 \ 0 \ 0 \rangle$ ,

$$f_2(x) = \langle 0 \ 3 \ 1 \ 0 \rangle, f_3(x) = \langle 0 \ 0 \ 1 \ 0 \rangle, \\ f_4(x) = \langle 0 \ 1 \ 2 \ 0 \rangle.$$

$k$  and  $K$ . The function is increasing and has a slope  $a \cdot k$  where it is positive. The output value is truncated by the mirror however and can not become negative. The mathematical representation is

$$f_I(x) = -k \text{ bndn}(K + a \cdot x) \quad (3)$$

The cost  $Q$  is the total number of transistors in the circuit, plus the weight of the input signal:  $Q = a + k + K + 1$ .

circuit structure II. With two mirrors, there are several possible structures. Summation of two mirror outputs always yields two circuits of type I, and is therefore uninteresting. A cascade of two mirrors is shown in fig. 9.a. This is a powerful circuit with five parameters. Actually circuit structure I can be obtained by setting  $k=0$  in structure II. The function  $f_{II}(x)$  has a transition point at  $x = K/a$ . For  $x < K/a$ ,  $f$  is constant or increasing with slope  $1 \cdot b$ , and for  $x > K/a$ , the slope is  $1 \cdot (b - a \cdot k)$ , as long as  $f_{II}(x)$  is

TABLE II. COST TABLE for Quaternary CMCL functions with the circuits of figs. 8 and 9 (circ. a) resp b)

no.	Q	f(x)	circ.	parameters
00	1	$\langle 1 \ 1 \ 1 \ 1 \rangle$		
01	3	$\langle 0 \ 1 \ 2 \ 3 \rangle$	a)	$K=0 \quad k=1 \quad a=1$
02	4	$\langle 0 \ 0 \ 1 \ 2 \rangle$	a)	$K=1 \quad k=1 \quad a=1$
03	5	$\langle 0 \ 0 \ 0 \ 1 \rangle$	a)	$K=2 \quad k=1 \quad a=1$
04	6	$\langle 1 \ 0 \ 0 \ 0 \rangle$	b)	$K=0 \quad L=-1 \quad k=1 \quad l=1 \quad a=1 \quad b=0$
05	7	$\langle 1 \ 1 \ 0 \ 0 \rangle$	b)	$K=1 \quad L=-1 \quad k=1 \quad l=1 \quad a=1 \quad b=0$
06	7	$\langle 2 \ 1 \ 0 \ 0 \rangle$	b)	$K=0 \quad L=-2 \quad k=1 \quad l=1 \quad a=1 \quad b=0$
07	7	$\langle 0 \ 1 \ 1 \ 1 \rangle$	b)	$K=1 \quad L=0 \quad k=1 \quad l=1 \quad a=1 \quad b=1$
08	8	$\langle 1 \ 1 \ 1 \ 0 \rangle$	b)	$K=2 \quad L=-1 \quad k=1 \quad l=1 \quad a=1 \quad b=0$
09	8	$\langle 2 \ 2 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=-2 \quad k=1 \quad l=1 \quad a=1 \quad b=0$
10	8	$\langle 0 \ 1 \ 2 \ 2 \rangle$	b)	$K=2 \quad L=0 \quad k=1 \quad l=1 \quad a=1 \quad b=1$
11	8	$\langle 0 \ 1 \ 0 \ 0 \rangle$	b)	$K=1 \quad L=0 \quad k=2 \quad l=1 \quad a=1 \quad b=1$
12	9	$\langle 3 \ 1 \ 0 \ 0 \rangle$	b)	$K=0 \quad L=-3 \quad k=2 \quad l=1 \quad a=1 \quad b=0$
13	9	$\langle -3 \ 3 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=-3 \quad k=2 \quad l=1 \quad a=1 \quad b=0$
14	9	$\langle 0 \ 0 \ 1 \ 1 \rangle$	b)	$K=2 \quad L=1 \quad k=1 \quad l=1 \quad a=1 \quad b=1$
15	9	$\langle 0 \ 1 \ 2 \ 1 \rangle$	b)	$K=2 \quad L=0 \quad k=2 \quad l=1 \quad a=1 \quad b=1$
16	9	$\langle 1 \ 2 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=-1 \quad k=2 \quad l=1 \quad a=1 \quad b=1$
17	10	$\langle 0 \ 1 \ 1 \ 0 \rangle$	b)	$K=3 \quad L=0 \quad k=1 \quad l=1 \quad a=2 \quad b=1$
18	10	$\langle 0 \ 0 \ 1 \ 0 \rangle$	b)	$K=2 \quad L=1 \quad k=2 \quad l=1 \quad a=1 \quad b=1$
19	10	$\langle 0 \ 1 \ 2 \ 0 \rangle$	b)	$K=2 \quad L=0 \quad k=3 \quad l=1 \quad a=1 \quad b=1$
20	10	$\langle 0 \ 2 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=0 \quad k=3 \quad l=1 \quad a=1 \quad b=2$
21	10	$\langle 1 \ 2 \ 0 \ 0 \rangle$	b)	$K=1 \quad L=-1 \quad k=3 \quad l=1 \quad a=1 \quad b=1$
22	11	$\langle 0 \ 1 \ 3 \ 3 \rangle$	b)	$K=2 \quad L=1 \quad k=2 \quad l=1 \quad a=1 \quad b=2$
23	11	$\langle 2 \ 3 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=-2 \quad k=3 \quad l=1 \quad a=1 \quad b=1$
24	12	$\langle 0 \ 0 \ 1 \ 3 \rangle$	b)	$K=1 \quad L=4 \quad k=1 \quad l=1 \quad a=1 \quad b=3$
25	12	$\langle 0 \ 1 \ 3 \ 2 \rangle$	b)	$K=2 \quad L=1 \quad k=3 \quad l=1 \quad a=1 \quad b=2$
26	12	$\langle 0 \ 3 \ 2 \ 1 \rangle$	b)	$K=1 \quad L=0 \quad k=4 \quad l=1 \quad a=1 \quad b=3$
27	12	$\langle 1 \ 2 \ 3 \ 0 \rangle$	b)	$K=2 \quad L=-1 \quad k=4 \quad l=1 \quad a=1 \quad b=1$
28	12	$\langle 1 \ 3 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=-1 \quad k=4 \quad l=1 \quad a=1 \quad b=2$
29	13	$\langle 0 \ 0 \ 2 \ 1 \rangle$	b)	$K=2 \quad L=2 \quad k=3 \quad l=1 \quad a=1 \quad b=2$
30	13	$\langle 0 \ 1 \ 3 \ 1 \rangle$	b)	$K=2 \quad L=1 \quad k=4 \quad l=1 \quad a=1 \quad b=2$
31	13	$\langle 0 \ 3 \ 1 \ 0 \rangle$	b)	$K=1 \quad L=0 \quad k=5 \quad l=1 \quad a=1 \quad b=3$
32	13	$\langle 1 \ 3 \ 0 \ 0 \rangle$	b)	$K=1 \quad L=-1 \quad k=5 \quad l=1 \quad a=1 \quad b=2$
33	13	$\langle 0 \ 2 \ 2 \ 1 \rangle$	b)	$K=4 \quad L=0 \quad k=1 \quad l=1 \quad a=3 \quad b=2$
34	14	$\langle 0 \ 1 \ 3 \ 0 \rangle$	b)	$K=2 \quad L=1 \quad k=5 \quad l=1 \quad a=1 \quad b=2$
35	14	$\langle 1 \ 2 \ 2 \ 0 \rangle$	b)	$K=5 \quad L=-1 \quad k=1 \quad l=1 \quad a=3 \quad b=1$
36	17	$\langle 0 \ 0 \ 3 \ 1 \rangle$	b)	$K=2 \quad L=3 \quad k=5 \quad l=1 \quad a=1 \quad b=3$

positive. Just as for circuit structure I, the output signal can not become negative due to the logic behavior of the second mirror. A range of functions is obtained that includes all step-up, step-down and delta literals, as well as a large number of other functions. A few examples are shown in figure 9.b. The mathematical representation is

$$f_{II}(x) = -1 \text{ bndn}(L + b \cdot x - k \text{ bndn}(K + a \cdot x)) \quad (4)$$

The cost  $Q$  is  $Q = a + b + k + l + K + L + 2$ .

With circuit structures I and II, a logically complete and powerful cost table can be designed. Structures with three mirrors have also been considered. All quaternary functions obtained, however, can also be obtained by a parallel combination of at most two circuits of the types I and II. Moreover, a cascade of three mirrors forms a long delay path and degrades the speed performance of the whole circuit. For these reasons, structures containing three or more mirrors will not be considered any further.

The cost table is presented in Table II. The functions in the table are ordered according to the cost  $Q$ . The table also lists the structure type and the values of the parameters.

For the decomposition program, it is necessary to know how the transistor counts of the cost-table functions and the mathematical cost functions are



Table III. The relations between the transistor count Q and the cost functions T.T.S., SUM, B.C. and T.C. for the costtable functions in table II

Q	T.T.S.						SUM							B.C.			T.C.		total
	1	2	3	4	5	6	1	2	3	4	5	6	7	0	1	2	0	1	
3			1										1				1		1
4		1							1					1			1		1
5		1						1						1			1		1
6		1						1						1			1		1
7		1	1	2				1	2	1				2	1	1	2	2	4
8		4						1	1	2				1	2	1	1	3	4
9		1	3	1				1	3		1			1	1	3	1	4	5
10		2	1	1				1	1	2				3	1		4		4
11			1	1							1	1		1		1	1	2	
12			1	2						1	2			2	1		3		3
13			2	2	1				1	2	2			4	1		5		5
14			1		1				1	1				2		2	2		2
17				1					1					1		1	1		1
total	3	9	12	4	4	2	4	3	7	8	6	4	2	8	17	9	8	25	34

related to each other. Table III lists the relations between the TTS, SUM, BC and TC cost functions and the transistor count Q for the functions in the cost table. The following observations can be made:

- For all cost-table functions,  $TTS \leq 6$ ,  $SUM \leq 7$ ,  $BC \leq 2$  and  $TC \leq 1$ .
- The correlation between the cost Q and the mathematical cost functions is weak. It is strongest for the total transition size and the transition count.

Because of this weak correlation, the mathematical cost functions are not useful to guide the decomposition procedure, which is explained in the next section. Instead, the properties of the cost-table functions mentioned in the previous paragraphs are used for this purpose.

#### 4.3. The cost-table decomposition program.

##### 4.3.1. Decomposition algorithms.

In the literature on cost-table decomposition of unary MVL functions, several decomposition algorithms have been proposed. The original method of Kerkhoff and Robroek [8] employs an exhaustive search over all possible combinations of cost-table functions. The minimum cost decomposition is guaranteed to be found, but the search requires much computation time. The method proposed by Lee and Butler [9] classifies the functions according to the transition count in order to guide the decomposition. In this way the number of combinations of cost-table functions to be considered is greatly reduced. Abd-El-Barr, Vranesic and Zaky [11] propose a different method to split a function, that exploits the specific features of CCD circuits, and also use the break count instead of the transition count.

Both the methods of Lee et al. and of Abd-El-Barr et al. are not directly applicable to general cost-table decomposition problems, because they are specifically designed to optimize CCD circuits. Schueller, Tirumalai and Butler [10] have considered the general problem of composing minimal cost tables of any size and they have done this for a variety of cost criteria. Their main conclusion is that minimal cost tables designed for various cost functions tend to contain the same functions.

A method similar to Abd-El-Barr's, which appears to be attractive in the case of CCD's, has been developed for CMCL circuits.

##### 4.3.2. Application of the modified decomposition method to CMCL circuits.

When adapting a method intended for CCD synthesis for CMCL circuits, the relevant features of both technologies must be compared. The CCD properties upon which the Abd-El-Barr method is based are the following:

- addition is inexpensive
- logic 3's are expensive to manipulate in CCD's, and therefore are always split.
- the break count is a good cost measure for CCD's, because every break in the function corresponds to an inhibit (which is relatively expensive) in the circuit.

In CMCL, addition is an inexpensive operation as well. However, 3's are not much more expensive to handle than other values, so the splitting procedure has to be changed. Also, the inhibit is not a basic element of CMCL, and therefore the break count is not directly related to the cost of CMCL circuits (refer to table III). Instead of using the BC or another mathematical cost function as a decomposition guideline, a function is checked against the properties explained in section 4.2.3 before it is looked up in the cost table. In this way, also a reduction in the number of lookups is achieved.

In the method of Abd-El-Barr, the functions resulting from the decomposition procedure are not looked up in a table. In the present design system however, all functions must be decomposed into cost-table functions. If the cost table does not contain the functions that result from the decomposition, the program must continue with other splits of the target function and if necessary of the subfunctions. This leads to a recursive algorithm. Special functions and function transformation are not considered because they can not be implemented in the general structure proposed in section 4.2. Because the complementing output structure (figure 8.c.) is used, the target function has to be transformed prior to the decomposition according to section 4.2.2. This is only done if the target function is not the multiple of a costtable function.

The adapted algorithm has been implemented in a computer program, CMCL\_SYN. The input of CMCL\_SYN is the function to be synthesized. The result is a decomposition based on the general structure and the cost-table functions presented in this chapter. A logic simulator for CMCL circuits designed with the basic elements of chapter 2 has been incorporated to verify the results of the decomposition program.

##### 4.4. Automatic Generation of the Layout.

In this section, the path from the synthesis results to the actual I.C. mask layout is described. The layout of the CMCL functions is generated by using standard-cell IC design tools made available through the NELSI project [19]. An overview of the design path is shown in figure 10.

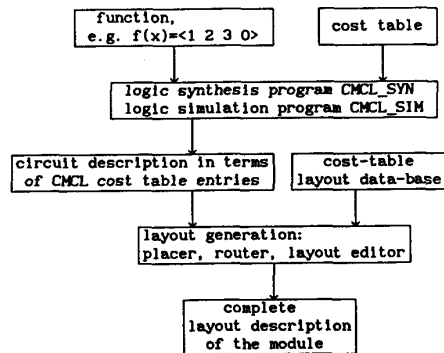


Fig. 10. Overview of the design tools.

The manually designed layouts of the cost-table functions are stored as cells in a layout library and in a cell description library. Figure 11 shows the mask layout and description for the cost-table function  $f(x) = \langle 1 \ 3 \ 1 \ 0 \rangle$ . A background layer defining the ground and power supply lines is also provided. The placer, router and layout editor together produce the mask layout description of the synthesized CMCL circuit.

The placer and router have originally been intended for gate-array design. It should be noted that the restrictions they impose result in a layout that is not as dense as could be obtained by a fully manual design. The advantages of automated design are bought at the expense of increased area consumption.

#### 4.5. An example.

As an example, the function  $f(x) = \langle 1 \ 3 \ 0 \ 1 \rangle$  is decomposed by CMCL\_SYN into the cost-table functions  $\langle 1 \ 0 \ 0 \ 0 \rangle$  and  $\langle 0 \ 0 \ 1 \ 0 \rangle$  with multiplication factors 2 and 3 respectively, summed and complemented with respect to the constant 3 (fig. 12.a). The resulting layout is shown in figure 12.b.

#### 5. Conclusions and suggestions for further research

A system for the automatic structural logic synthesis and layout generation for unary multiple-valued current-mode CMOS logic circuits has been presented. The logical synthesis part is based on modified cost-table decomposition techniques. The layout generation is carried out using standard cell placement and routing tools. The system has been developed for quaternary unary functions, but will be extended to functions of more variables.

The system presented here is intended as a module generator for combinational logic circuits, as part of the I.C. design environment. The resulting layouts can be used as modules in binary VLSI integrated circuits.

The research in this area continues. The following area's will be the subject of further research:

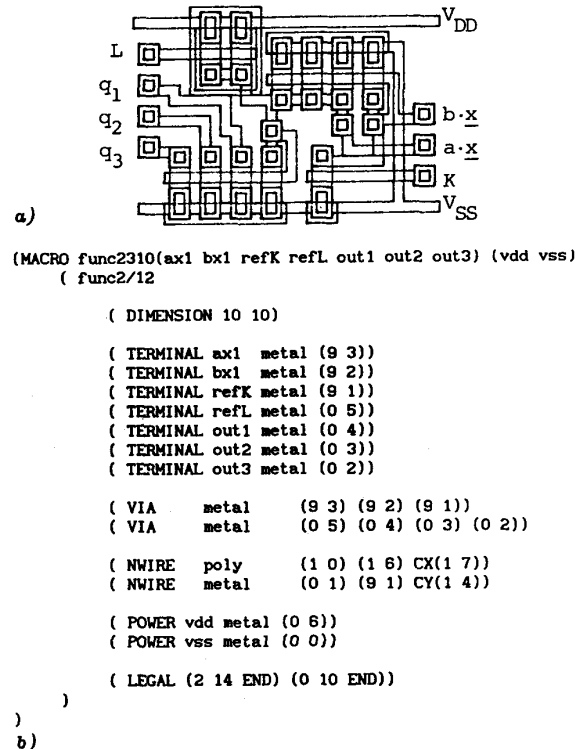


Fig. 11. Representation of cost-table functions for the placer and router programs: a) layout; b) layout description.

- preparation of optimal CMCL cost tables of a given size
- circuit synthesis based on the Davio/Deschamps method
- comparing the results with other synthesis techniques
- comparing the synthesis results with binary circuits
- using more flexible layout generation tools
- investigation of error generation and propagation
- investigation of sequential CMCL circuits
- the interface with binary circuits.

#### Acknowledgements.

The authors wish to thank prof. O. W. Memelink for help and encouragement. Also thanks are due to H. Banning and V. Rikkink for programming and layout design.

The layout software used was developed at the universities of Delft and Eindhoven and by G. Piepers at Twente University.

These investigations in the program of the Foundation for Fundamental Research on Matter (FOM) have been supported by the Foundation for Technical Science (STW), a division of the Netherlands Organization for the Advancement of Pure Scientific Research (ZWO). This research was partially funded by Nato grant no. 423/84.

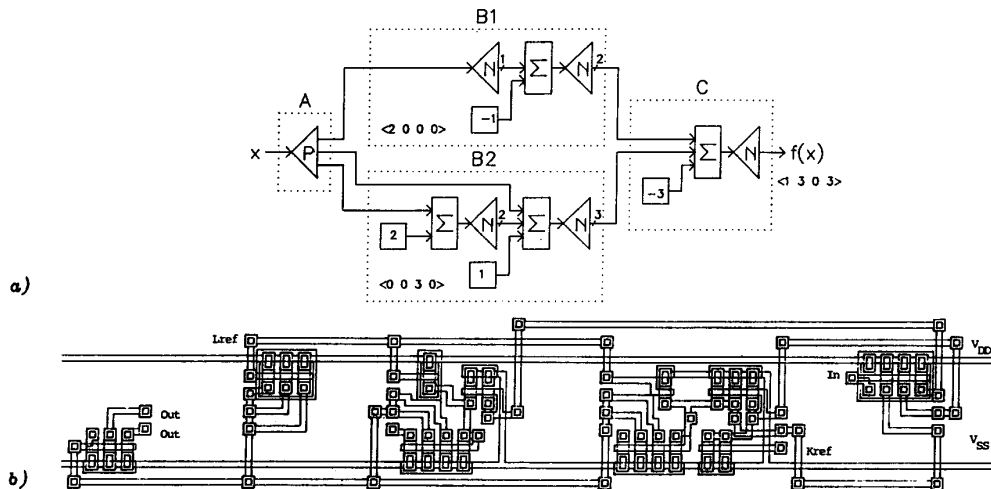


Fig. 12. Example of the design results: a) decomposition of the function  $f(x) = \langle 1 \ 3 \ 0 \ 3 \rangle$ , b) the automatically generated layout of this function.

#### REFERENCES.

- [1] Freitas, D.A. and K.W. Current, "A Quaternary Logic Encoder-Decoder Circuit Design using CMOS," in *Proc. 1985 ISMVL*, pp. 84-90, May 1985.
- [2] Yamakawa, T., "CMOS Multivalued Circuits in Hybrid Mode," in *Proc. 1985 ISMVL*, pp. 144-151, May 1985.
- [3] Yamakawa, T., T. Miki, and F. Ueno, "The Design and Fabrication of the Current-Mode Fuzzy Logic Semi-Custom IC in the Standard CMOS IC Technology," in *Proc. 1985 ISMVL*, pp. 76-82, May 1985.
- [4] Onneweer, S.P. and H.G. Kerkhoff, "Current-Mode High-Radix Circuits," in *Proc. 1986 ISMVL*, pp. 60-69, Blacksburg, May 1986.
- [5] Onneweer, S.P. and H.G. Kerkhoff, "High-Radix Current-mode CMOS Circuits Based on the Truncated-Difference operator," in *Proc. 1987 ISMVL*, pp. 188-195, Washington, D.C., May 1987.
- [6] Kawahito, S., M. Kameyama, and T. Higuchi, "VLSI-Oriented Bi-Directional Current-Mode Arithmetic Circuits Based on the Radix-4 Signed-Digit Number System," in *Proc. 1986 ISMVL*, pp. 70-77, Washington, DC, May 1986.
- [7] Kawahito, T.S., M. Kameyama, T. Higuchi, and H. Yamada, "A High-Speed Compact Multiplier Based on Multiple-Valued Bi-directional Current-mode Circuits," in *Proc. 1987 ISMVL*, pp. 172-180, Washington, D.C., May 1987.
- [8] Kerkhoff, H.G. and H.A.J. Robroek, "The Logic Design of Multiple-Valued Logic Functions using Charge-Coupled Devices," in *Proc. 1982 ISMVL*, pp. 35-44, May 1982.
- [9] Lee, J. and J.T. Butler, "Tabular Methods for the Design of CCD Multiple-Valued Circuits," in *Proc. 1983 ISMVL*, pp. 162-170, May 1983.
- [10] Schueller, K.A., P.P. Tirumalai, and J.T. Butler, "An Analysis of the Costtable Approach to the Design of Multiple-Valued Circuits," in *Proc. 1986 ISMVL*, pp. 42-50, Washington, DC, May 1986.
- [11] Abd-el-Barr, M.H., Z.G. Vranesic, and S.G. Zaky, "Synthesis of MVL-Functions for CCD-Implementations," in *Proc. 1986 ISMVL*, pp. 116-127, Washington, DC, May 1986.
- [12] Kerkhoff, H.G., *Theory, Design and Applications of Digital Charge-Coupled Devices*, H. G. Kerkhoff, Enschede, 1984.
- [13] Post, E.L., "Introduction to a General Theory of Elementary Propositions," *American Journal of Mathematics*, vol. 43, pp. 163-185, Dec. 1921.
- [14] McCluskey, E.J., "Logic Design of Multivalued IIL Logic Circuits," *IEEE Tr. Comp.*, vol. c-28, no. 8, pp. 546-559, August 1979.
- [15] Allen, C.M. and D.D. Givone, "A Minimization Technique for Multiple-Valued Logic Systems," *IEEE Tr. Comp.*, pp. 182-184, Feb 1968.
- [16] Vranesic, Z.G., E.S. Lee, and K.C. Smith, "A Many-valued Algebra for Switching Systems," *IEEE Tr. Comp.*, vol. c-19, no. 10, pp. 964-971, Oct. 1970.
- [17] Su, S.Y.H. and A. A. Sarris, "The Relationship between Multivalued Switching Algebra and Boolean Algebra under different definitions of Complement," *IEEE Tr. Comp.*, vol. c-21, no. 5, pp. 479-485, May 1972.
- [18] Davio, M. and J.-P. Deschamps, "Synthesis of Discrete Functions Using IIL Technology," *IEEE Tr. Comp.*, vol. c-30, no. 9, pp. 653-661, September 1981.
- [19] Dewilde, P., *The Integrated Circuit Design Book*, Delft University Press, Delft, 1986.